EXHIBIT F
PART 1

RealSecure™ 1.2

User Guide and Reference Manual

Chapters:

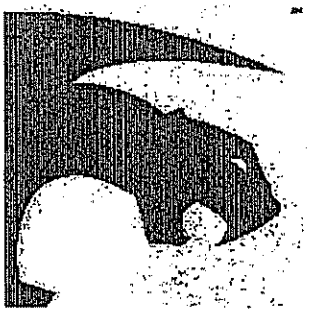
Appendix:

Copyright © 1996, 1997 Internet Security Systems, Inc. All Rights Reserved.

Technical Support: rs-support@iss.net

Table of Contents

Appendix A: Features and Attack Signatures



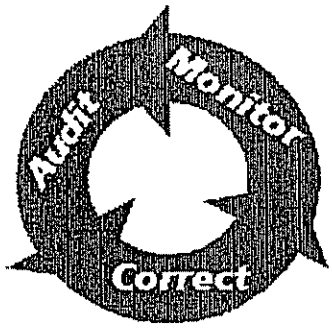
RealSecure™

Chapter 1:

Introduction to Real-Time Intrusion Detection

OVERVIEW

Security is a hot topic today, especially on the Internet. Very few people understand how to achieve an acceptably secure system. Some companies sell products that promise to make a network secure. What they neglect to mention is... "there is no single solution to security". Security is a process and method of doing business that requires continual updating.



The Security Cycle

Audit

The illustration above depicts the cycle used to dynamically improve the security of a system. An auditing tool is used to find the holes in a network. Combining a security audit with a security policy establishes the original baseline security.

Correct

The parts of the system which failed the audit are corrected. Subsequently, the system needs to be audited again and again to ensure that new holes do not appear or old ones resurface.

Monitor

A monitoring tool is used to watch for new security breaches or attempts to abuse old holes. This keeps the security manager in touch with the state of the network.

RealSecure... falls into the third category. It resides on a computer connected to the network and watches the network traffic. This allows it to compare traffic to a wide variety of attack signatures. It summarizes the information in a concise manner so the security manager can understand what is happening on the network, *as it happens*.

Common Uses for Network Intrusion Detection

RealSecure is very versatile and therefore performs many different functions. Some possible uses include:

- Auditing network utilization. (How many hits does each Web server get? Who is transferring files from the FTP site?)
- Auditing and blocking network intrusion attempts. (How many attacks are from non-US sites? How many rlogin attempts are made and from where?)
- Providing a second tier of security behind a firewall. (There should never be telnet sessions incoming to the internal network. So, kill and log any such attempts.)
- Detecting network state and possible problems. (Did Sales just configure a new machine on the network? Who is using it? Was an IP address of an existing machine used?)
- Obtaining profiles of a detected intruder and assessing damage. (So, perhaps it is known that the intruder is using the Web server to attack other sites. In this case, consider logging all of his/her session data throughout the night and replay it later.)

Remember, **RealSecure** can analyze **any** network traffic. When watching the network for invalid login attempts, **RealSecure** prints all invalid login attempts made over the network. This is a great improvement over normal auditing in which the administrator has to search a log from each system on the network for suspicious activity.

Dangers

Because **RealSecure** watches and responds to network events, there are some associated risks. First, it can log all data in a connection including keystrokes and e-mail. Exercise discretion when using this feature. Once logs are created, they should be kept on a secure host as they may contain passwords or other sensitive data. This is one reason to run **RealSecure** on a dedicated machine. Intruders will perceive the monitoring machine

as a prime target, both to steal its saved data and to erase evidence of their actions.

Second, **RealSecure** responds to events. In particular, the 'kill' option, if misused, can block traffic over an entire network. Ensure that any connections being killed are ones that require blocking. A wildcard rule with a 'kill' action will block all connections, including http, telnet, and ftp.

The rule of thumb to use is "think twice, configure once".

Legality

In most cases, the United States Government has upheld the right of individuals to monitor their own networks. The general consensus is to notify all users of monitoring. Consult a lawyer if there are any questions as to the legality of this product's use. For explanation of legal issues, refer to the CERT advisory provided at the back of this manual.

(This page intentionally left blank)

Chapter 2:

Installing RealSecure™

OVERVIEW

To obtain maximum benefits, install RealSecure™ on a dedicated machine at an entry point to the network. Good places to consider would be at the Ethernet interface just inside the firewall or between the Internet router and the internal machines. For security and performance, it is strongly recommended that RealSecure be run on its own dedicated machine (ideally, the machine should have as many of its own services as possible disabled). Also, the only user(s) should be the administrator(s). RealSecure collects a lot of data from the network, so the more power and disk space the machine has, the better.

This chapter covers the following topics:

- System Requirements
- Installation Instructions

System Requirements

The **RealSecure** engine requires the following configuration:

- SunOS 4.1.x, Solaris 2.3 and up, or Linux (kernel versions 1.3.x and later).
- An Ethernet interface connected to the target network.
- 486-class machine or better; however, for non-dedicated machines running **RealSecure**, more resources will be necessary.
- A minimum of 25 MB of available disk space.
- A minimum of 16 MB RAM, 32 MB recommended.

The **RealSecure** GUI requires the following configuration:

- The X-Window system, version 11 or later.
- Motif Installation (Solaris 2.3 and 2.4 only).
- A minimum of 32 MB RAM.

How to Install RealSecure

Key Processing

To activate **RealSecure** obtain a license key file (iss.key). To obtain a license key file, please contact ISS immediately at (770) 395-0150 or by E-mail at keys@iss.net. Upon receiving a license key, save the key as the filename iss.key in the /usr/local directory.

Old versions of the software (Release 1.0) will continue to operate using the old key, but to utilize the new functionality of **RealSecure**, obtain a new key from ISS. To obtain a new key, e-mail the existing key along with name and address of the organization and

the e-mail address where the new key is to be sent to keys@iss.net. ISS will update key processing records and forward the new key promptly.

Note: Save the e-mail message as "iss.key" in your /usr/local directory. It is not necessary to decode this key. Save the entire message or just the key. The key MUST include the "BEGIN" and "END" lines.

For further questions about key processing, contact support@iss.net.

Step 1: Obtain the Distribution Software

Installing from the ISS Web site:

- To download **RealSecure** from ISS, access:

  http://www.iss.net/RealSecure from within a Web browser.

Step 2: Copying the Distribution Software to the Destination System

- The tar file must be loaded to each machine that will be running either the **RealSecure** GUI or engine. Methods to transfer the archive include FTP and e-mail.
- Perform the remaining steps on *each* machine running **RealSecure**.

Step 3: Untar the Archive and Run the Install Program

- Change to the directory the directory where the tar file resides, for example:
  # cd /usr/local

- Next, enter the following commands:
  # tar xvf rs-*.tar

  # cd rs

  # ./install.rs

Step 4: Start sssd on Each System Running the RealSecure Engine

Note: Perform Step 4 only for multiple engines.

- To run the **RealSecure** engine, install sssd on the host.
- To run the **RealSecure** engine remotely, or to run multiple engines in order to sniff multiple segments or networks, perform these tasks for each machine for which the **RealSecure** GUI will run.

Installing from CD-ROM:

- Mount the CD-ROM volume (refer to the UNIX manual for instructions).
- From the directory where the CD-ROM is mounted, perform the following:

```
./install.rs
```

A. How to Set Up the sssd Engine

The engine and **RealSecure** GUI setup can be automated by running sss-setup. sssd engines require an authentication file be created to restrict access to authorized hosts.

To manually edit the file, perform the following procedure:

- Log in as root on the machine on which the engine will run.

- Create a file containing the name or IP address of each machine on the network where the **RealSecure** GUI is being installed. Indicate a random pass phrase used to authenticate the connection.

  This authentication file specifies allowable hosts for connectivity to sssd, in order to start up engines and to indicate a pass phrase used during connections for authentication and encryption. The default location of this file for the sssd server is /etc/sssd.auth. To change the default, specify the -a option along with an alternate pathname for the authentication file on the command line when running sssd.

  The format of the authentication file is:

  <Pass Phrase>//<hostname>

  <hostname> can be a domain name or IP address. <Pass Phrase> should be a unique, hard-to-guess set of letters, numbers and punctuation. Do **not** use a single word, sentence or sentence fragment from a well-known published book or song, or one that is guessable. The sssd server runs an engine capable of monitoring all network traffic. Access through the sssd server poses serious compromise to the security of a system. Because IP addresses can be spoofed, it is vital to system security to choose a good pass phrase. The authentication file is owned by root and must **not** be readable by users on the system other than root. Be sure to create the file with proper modes **before** entering pass phrases. Take a look at the example on the following page:

```
# whoami

 root

 # cd /etc
 # echo > sssd.auth
 # chmod 600 sssd.auth
 # echo ad98IU Ai2 ah c89 kqaknsdh//mvquibox.mvdomain.com >
```

B. Set Up the GUI

To manually setup the GUI, perform the following procedure:

Run **RealSecure** on the machine from which the **RealSecure** GUI will run. Next, create an authentication file containing the hostname and pass phrases of each host running sssd for which **RealSecure** will run. This includes the local host when only planning to run local engines.

The default filename for this file is /etc/sss.auth. Match the entries in this file to the pass phrase in the /etc/sssd.auth file on each machine. If running only a local engine, accomplish this by simply copying the sssd.auth file to sss.auth. As with the sssd.auth file, be sure no one other than root can read the contents of the file.

Because the machine running the **RealSecure** GUI controls the remote engines on all the machines in the sss.auth file, it is extremely important the machine remain secure against attack. If at all possible, run the GUI on a machine having no untrusted users and no services running. Take a look at the following example:

```
# whoami

 root
```

```
# cd /etc
# echo ad98IU Aj2 ah c89
kqaknsdh//ssdbox1.mydomain.com > sss.auth

# echo okPui uz 472 JK cnzx
opzutb//sssdbox2.mydomain.com >> sss.auth
```

C. How to Start sssd

To start the sssd daemon from a boot startup file, place the following command in a system rc file the system uses (i.e.,/etc/rc, /etc/rc.local, or /etc/rc.2/SXX.sss, etc.). The file location depends on the operating system:

```
sh -c "(cd <RealSecure_distrib_directory>; ./sssd -s )"
```

Where: <RealSecure_distrib_directory> is the directory path in which the **RealSecure** engine is installed.

Start sssd from a shell command prompt by entering the same command provided above while logged in as root. For verbose output, run the foreground task using the -f option and use the -v option along with -f.

The following summarizes sssd command line options:

-a <file>	Authentication file and path (default: /etc/sssd.auth)
-f	Foreground processing (do not run as daemon)
-p <port>	Port number to listen for connections
-s	Log connections to syslog
-v	Verbose output

How to Stop sssd

Stop the sssd daemon with the kill command, for example:

```
# kill -TERM `ps ax | grep sssd | grep -v grep | cut -f1 -d" "`
```

Arguments to ps may vary.

Note: It is not necessary to start the sssd server if only running an engine on the local machine.

Step 5: How to Start RealSecure GUI (rsgui) on the Administrative Machine

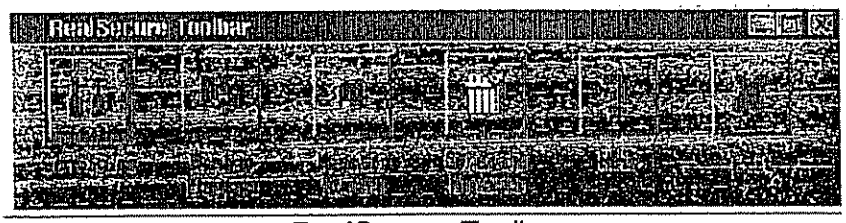
Enter the following commands:

```
# cd rs

# ./rsgui
```

Note: Upon entering the rsgui command, an About box displays and will eventually time out if the OK button is not pressed.

The **RealSecure** toolbar displays:

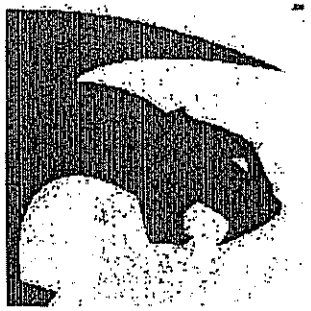


***RealSecure** Toolbar*

The toolbar allows for:

- General Configuration
- Configuring **RealSecure** engines
- Viewing network events
- Generating reports
- Help
- Exit

(This page intentionally left blank)



Chapter 3:

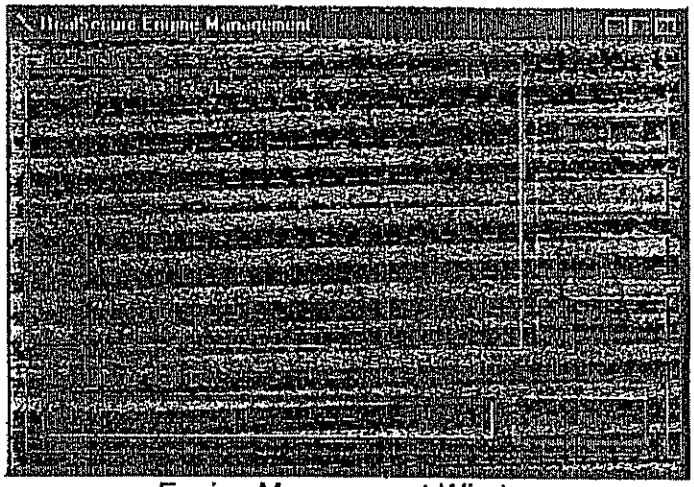
Configuring RealSecure

OVERVIEW

To effectively monitor network problems, provide RealSecure with important network configuration requirements. For instance, a Web server should be having Web traffic, but the company's accounting machine probably should not. In this case, consider configuring RealSecure to ignore Web traffic to the Web server, while continuing to log all other Web traffic. Each service accessed through the network in the same manner.

RealSecure has two related, but separate configuration modes. The filter configuration mode sets what services **RealSecure** watches for connections. The feature configuration mode enables, disables and fine-tunes custom attack signatures.

To set GUI configurations, access the Engine Management window. To display this window, choose Configure Engines from the **RealSecure** toolbar.



Engine Management Window

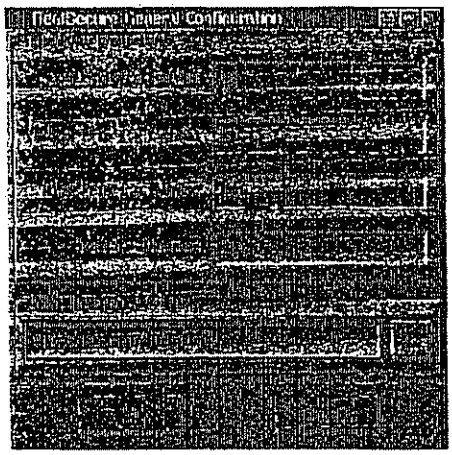
The **RealSecure** Engine Management screens allow for configuration and monitoring of the state of **RealSecure** engines on the network. The active engines window displays a list of all **RealSecure** engines currently communicating with the network's GUI. Each engine normally remains in an "Alive" state. The GUI periodically pings each engine to check for responsiveness. If it fails to respond to a ping, it moves into a "No Response" state, indicating the engine may no longer be communicating properly with the GUI.

At the bottom of the Engine Management window is a "Hostname or Address" field. Entering the name or IP address of an engine and then pressing Enter or clicking on "Start Engine" causes the

GUI to communicate with the "sssd" daemon running on the remote machine; this starts a **RealSecure** engine on that machine. For an engine to startup remotely, **RealSecure** must be installed and the authentication entries should be setup and running sssd on the remote

machine. To start up a local engine, use the hostname "localhost" or the IP address "127.0.0.1". Starting up local engines does not require sssd to be running or any authentication information to be configured.

By selecting a running engine, one can then click on any of the buttons (located on the right side of the screen) and perform the specified actions.



General Configuration window

The General Configuration screen allows for adjustment to of **RealSecure** timeouts and settings to fit a particular network configuration. Once settings are configured, additional changes to this screen are not necessary, unless the system or network configuration has also changed.

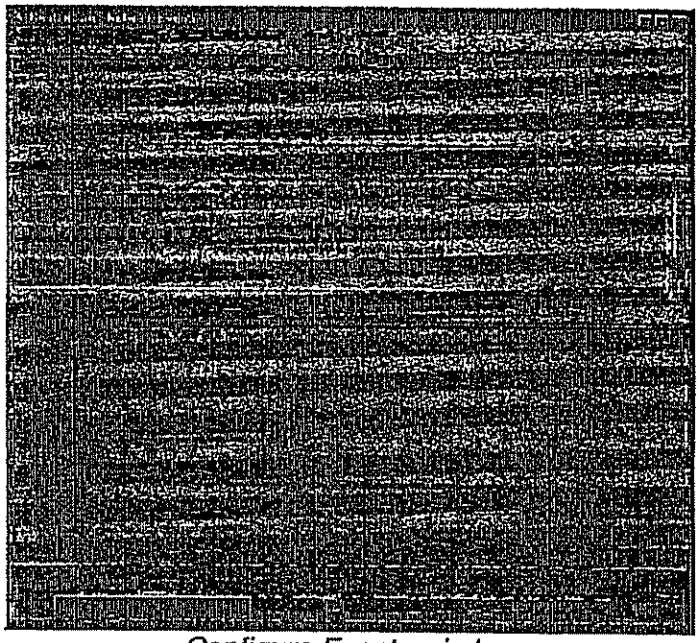
The three event timeout boxes allow for entering the number of seconds that events remain on the GUI screen and is monitored internally by the **RealSecure** engine. The higher these timeout values, the more memory the engine and GUI utilize to track all of the events. Regardless of these values, if logging for events is enabled, a permanent record of each event that occurs in the log files is kept.

The Inactive Streams timeout allows for setting a value for how long the TCP stream data is kept in memory before it is discarded. Decreasing this value will save memory, but if set too low, signatures that occur with a larger amount of time between packets may be interpreted separately instead of as a continuous stream.

The HTML browser location is the path of the Web browser. If a Web browser is in the path, there is no need to specify a full pathname, although it is recommended in case the path variable should change at some point. Clicking on the "..." icon displays a directory browse window from which the Web browser is found.

Filter Configuration

The filter configuration does exactly what its name implies; it selects the types of network connections **RealSecure** should ignore or watch. To edit the filter configuration, click on the desired engine for configuration in the Engine Management window and then click on **Filters**. The Configure Events window displays. Alternately, the filter.cfg file on the engine's host are editable.



Configure Events window

A filter consists of rules matching in order. Once a match is found, further searches cancel. If the engine being configured has a filter configuration, the rules appear in the Filter List. To modify a rule, click on the rule. It will appear in the bottom half of the window. Clicking on any of the fields allows for modification. Clicking on the **Modify** button commits changes to the list. To add a rule, click on the position in the list where the new rule is to be inserted. Enter the desired rule in the fields in the bottom half of the window. To add the new rule, click on the **Add** button.

There are several fields to a rule, they are as follows:

- Source and Destination Address -- IPs or ranges of IPs
- Source and Destination Service -- TCP/UDP ports
- Source and Destination Type -- ICMP only
- Protocol -- TCP, UDP, or ICMP
- Label -- a one-word description of the event that appears in the GUI
- Priority -- the severity of this rule
- Actions -- what to do when this rule is matched

Addresses

The source and destination addresses are structured in the common dotted decimal form (i.e., 10.1.2.3). To specify a range of addresses, use the asterisk (*) wildcard. For instance, an address of 10.1.1.* would match 10.1.1.2 and 10.1.1.50, but not 10.1.18.2. Wildcards must be on even boundaries. For example, 10.* is valid, but 10.1.1* is not. Finally, a wildcard by itself will match all addresses.

Services

Services are the ports in a connection. For instance, HTTP (Web) traffic uses port 80. To select a service, click on the button. A list of services appears. Select the desired service and click OK. Selecting the **Any (0)** service matches any service. If the desired

service is not in the list, edit the services file included with the distribution software to add the desired service.

ICMP Types

Every ICMP packet has a type and sub-type. For instance, ping packets have an ICMP type of Echo Request. To select a type, click on the button. A list of types appears. Select the desired type and click OK. If the desired service is not in the list, edit the services file included with the distribution software to add the desired ICMP type.

Protocols

Each rule must be of a specific protocol type. Valid protocols are **TCP**, **UDP** and **ICMP**. TCP is a reliable data transport used for services like E-Mail, FTP, and HTTP. UDP is a datagram service used for services like CU-SeeMe and Talk. Lastly, ICMP is used for sending control messages between Internet nodes.

Labels

The label is a one-word tag for this particular rule. It appears in the logs and on the display. It allows for differentiation of connections at a glance. Valid tags include Web-Traffic or Bob's_PC, but **not** My Server (note the space).

Priority

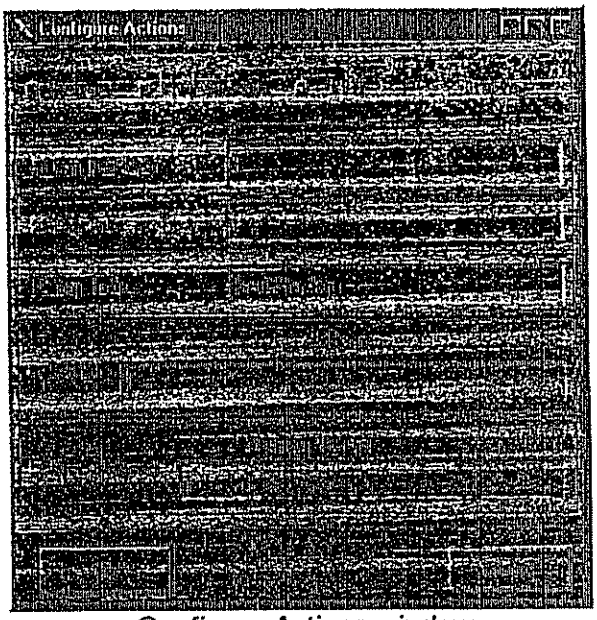
Each rule has a priority that controls which window the event appears and additional group events for generating reports. Valid priorities are **high**, **medium**, and **low**. It is best to group rules based on priority, to filter out common events (Web transfers, e-mail) from less common events (attempts to exploit security holes, connections to the accounting machine).

Actions

The action configuration is the same for both filters and attack pattern matching. There are quite a few possible actions, they are:

- Ignore any event matching this rule
- **D**isplay a message in the main window where the event occurred
- **V**iew the data from the connection in real-time
- **K**ill the connection by sending a reset packet (only possible with TCP connections)
- **M**ail a notification to the administrator
- Run a **U**ser-specified program when the event occurs
- Log data to a file:

- Log Info that the connection occurred to a file
- Log Text data sent through the connection
- Log Raw data sent through the connection for later playback



Configure Actions window

To enable or disable an action, click on it. When the **Ignore** action is enabled, all other actions are disabled. Some actions require additional data. For instance, if the **Mail** action is enabled, **RealSecure** must have an address in order to send mail. After selecting the desired action(s), click OK to continue.

Sample Uses of Filter Rules

Getting an Idea of Your Network Traffic

To initially become accustomed to filters and to become familiar with network traffic patterns, use filter.cfg (which is the default).

Be sure to log in as root. Follow the startup instructions for using **RealSecure**. This configuration displays all TCP and UDP connections on the network. The display can get very crowded, quickly. Therefore, close down any engines still running by selecting each engine in the Engine Management window and clicking on the Shutdown button. If any/all engines are to remain running after exiting the GUI, it is not necessary to shut them down from within this window. To exit the GUI, click on the Exit button located on the toolbar.

Review the network security policy. Is rlogin permitted from anywhere, or just from internal hosts? Examining the firewall's configuration is helpful. The entries in the filter.cfg file are matched sequentially (one-by-one). A match indicates the action specified at the end of the line is taken, and further matches are discarded. For this reason, save wildcard entries until the end.

Determine the filter rules based on the network security policy. Usually, these correspond with the firewall configuration. Thus, **RealSecure** is used as a second level of defense. **RealSecure** will recognize and display exactly all occurrences of unauthorized traffic.

To see a general configuration that shows all common services refer to the filter.cfg included by default in the rs directory.

Since **RealSecure** engines communicate with the GUI host via normal UDP packets, configure **RealSecure** to ignore them. Use the following rules to ignore all **RealSecure** reports destined for the GUI host:

```
udp 0.0.0.0/0 10.0.0.1/32 0 900 RSgui 3 ignore
udp 0.0.0.0/0 10.0.0.1/32 0 901 RSeng 3 ignore
```

Sample Configurations

To get started using **RealSecure**, three sample configurations are included. To use them, copy the sample filter.cfg and features.cfg to each of the hosts that will be using them. The following describes each of the three configurations:

- All filter/ features (default) - Enables all **RealSecure** checking and data decoding.
- Exp filter/ features - Enables only exploit checking (leaves out some of the general information features).
- Web filter/ features - Enables Web checks only.

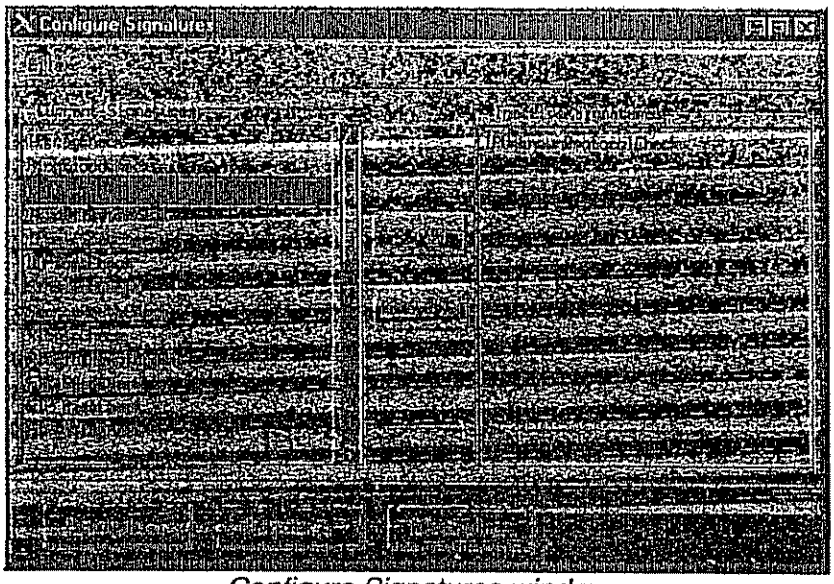
Feature Configuration

RealSecure has a standard set of attack signature checks that examine data inside a connection and check for significant signs of intrusion attempts. For instance, one of the old Sendmail exploits involved passing a malformed **From** address in an e-mail header. **RealSecure** watches all SMTP connections for this invalid data and triggers an alarm once it detects such an attack. The network administrator can then investigate and use the logging and response options of **RealSecure** to trace and ultimately lock out the intruder.

To enable signature checking, the filter configuration must have an entry for the service. For instance, Sendmail bug check is enabled (see above), but port 25 (Sendmail) is not being watched, subsequently the check is never used. To obtain more information about which filters are needed for each service, refer to the Appendix.

For a list of the checks and a definition of their capabilities, refer to Appendix A, **RealSecure** Features and Attack Signatures.

Each check has a standard set of options, configurable through the GUI. Select the engine to configure in the Engine Management window and then click on the **Signatures** button. The Configure Signatures window displays. Alternately, the features.cfg file is editable.



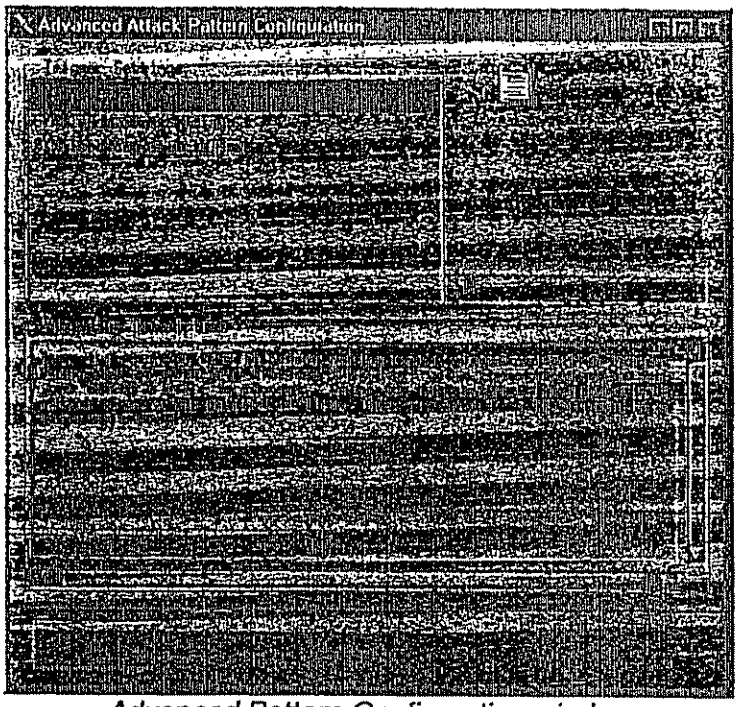
Configure Signatures window

Each check has a standard format scheme for its options. For instance, the check for IP fragmentation attacks contains the prefix **IPFrag**. To manually change the priority of this check, modify the entry called **IPFragPriority** in the features.cfg file.

The following details common options for all checks:

- ...Check - whether to perform this check (yes or no)
- ...Priority - the priority level for this attack
- ...SMessage - the message to log when this attack happens
- ...Actions - a list of actions to take when **RealSecure** sees this attack

Also, there are tunable parameters for some of the checks. To prevent false alarms, set the sensitivity of these checks. **RealSecure's** default configuration has reasonable values for these settings. These settings may vary, depending on the network. They are configurable by selecting the check in the Configure Signatures window and clicking on the **Advanced** button.



Advanced Pattern Configuration window

Checks

IP Fragmentation

This check looks for IP fragments with a size less than or equal to **IPFragThreshold**. Fragmentation requires that the data portion of the generated fragments (that is, everything excluding the IP header) be a multiple of 8 bytes for all fragments other than the final one. To check for an offset less than 16, use a setting of **2**. There is no need to change the default value.

Ping Flooding

This check determines if more than **PingFloodPackets** are received in **PingFloodDelta** seconds. The default setting is 50

packets in 3 seconds. If the network is on a slow connection like 14.4 PPP, consider making this setting more sensitive. Otherwise, the default value should suffice.

Arp Check

If someone attempts to contact a host that is powered down, multiple address request packets are sent with no response. **ArpMaxUnAcked** sets how many requests are sent to an unresponsive host, before triggering an alarm.

SYN flood Check

A SYN flood is a Denial of Service attack created by filling up the listen queue of a machine so that there is no room for legitimate users to establish a connection. If this attack is detected, and the "Kill" action is set, **RealSecure** will implement a random drop algorithm that frees up an entry in the listen queue for a legitimate connection.

In order to optimize the effectiveness of this algorithm, it is necessary to set the advanced parameter 'SYNFloodHighWaterMark'. This is the number of SYNs to allow to wait in each machine's queue for a response before the random drop algorithm is implemented. This number should be smaller than the size of the listen queue for your machines by some percentage. A guideline for this is 70% of the size of your listen queue, but this value will need to be fine-tuned to find what works best for each individual network.

The larger the size of the listen queue, the more effective **RealSecure** will be in allowing the queue to remain open at most points in time. Contact your vendor for information about how to increase the size of this queue, or to determine its current value.

Sample Configurations

The **RealSecure** sample directory contains several example configuration files that demonstrate various ways to use **RealSecure** on a network. Modification of these samples is not necessary. Next, set aside time to preview **RealSecure's** various features. Once you have explored each distinct option, consider creating a customized written configuration (refer to the files in the rs/sample directory for more details regarding customizing a configuration).

Each sample configuration contains two files, filter.cfg and features.cfg. The filter.cfg file determines which packets on the Network, **RealSecure** is to examine and suitable action(s) based on those packets. The features.cfg file contains information about how to analyze the traffic being examined in the filter configuration.

To use these samples, copy the sample filter.cfg and features.cfg to the main **RealSecure** GUI directory.

Sample Configuration 1: WebWatcher

web-features.cfg
web-filter.cfg

The WebWatcher configuration logs all URLs transmitted across a network to the **RealSecure** GUI and to an http.log file, while ignoring all other traffic. This allows for tracking and monitoring Web site traffic in real-time and stores Web page(s) logs for which user's access. The administrator should refer to the network security policy that limits the types of Web page(s) and sites' users are permitted to access. As well, consider restricting the hours in a day and allotted online time users are permitted to "Surf the Web."

This configuration allows for tracking and monitoring of how each user is utilizing the World Wide Web (WWW) within an organization; all the while, not compromising speed or accessibility to the WWW. Identify violations to the network security policy

either in real-time from the GUI or later by reviewing log files. Subsequently, take administrative action(s) based on the source and time of any such violation. Log source and time reports if necessary and use them to document network security violations as required.

This configuration produces logs that monitor Web traffic on the network Web server, without ever modifying it. Web traffic logs provide essential information that consequently administers a backup mechanism. These traffic logs both maintain and log all Web connections on the server; even if network disk space becomes full or logs become inadvertently deleted.

In the filter.cfg file, the source and destination addresses are set to the wildcard address, **0.0.0.0/0**. This file logs incoming connections from any source address to any destination address. If there is a specific machine installed to monitor outgoing Web connections, consider changing the source address value to the IP address of the machine; and only connections originating from that machine will log.

Ignore all outgoing traffic while monitoring the Web server. Modify the destination address to the IP address to that of the Web server. The source port in the configuration is set to 0, indicating connections coming from any source port will be logged. The Message value is set to Web. Priority is set to 2 (Medium). Connection establishment on the GUI is viewable real-time, since the Display action is enabled.

In the features.cfg file, the HTTPGetDecodeCheck is enabled, and all other checks disabled. The action for this check is set to Display and to log to a file.

Sample Configuration 2: Exploit Finder

exp-features.cfg
exp-filter.cfg

This configuration displays all attempts of known exploits against any machine on the network, or originating from any machine on the network. This immediately allows an administrator to check for a break-in, using one of the many ways **RealSecure** checks. **RealSecure** also detects if anyone inside the network is attempting to break-into other machines on the Internet.

To configure this, set up the features configuration to enable all of the exploit checks, while disabling all the simple informational decodes. The filter configuration is then set to filter out services the exploits are attempted against, in this case HTTP, SMTP, FTP, Rlogin, and Ident. Consider configuring the individual features to page or flag some other alert when detecting one of these exploits. These features will indicate a break-in attempt.

Sample Configuration 3: TCP Traffic

all-features.cfg
all-filter.cfg

This configuration displays all TCP traffic going through the network and displays all decodes and vulnerability attempts. This configuration is good for a small network having an active system administrator. Watching **RealSecure**, the administrator can monitor all of the TCP traffic on the network, users and passwords being attempted to each machine files being transferred over FTP and URLs being used over HTTP. These attempts are harder to detect on larger networks.

Changing the one line of the filter configuration to point to a specific machine, can significantly reduce the information that pinpoints all of the TCP traffic coming and going from one particular machine. If there is a suspected hacker on a machine, consider this option to monitor the traffic on the machine. If this is the case, consider enabling log in to a file in order to efficiently monitor the activities of the hacker.

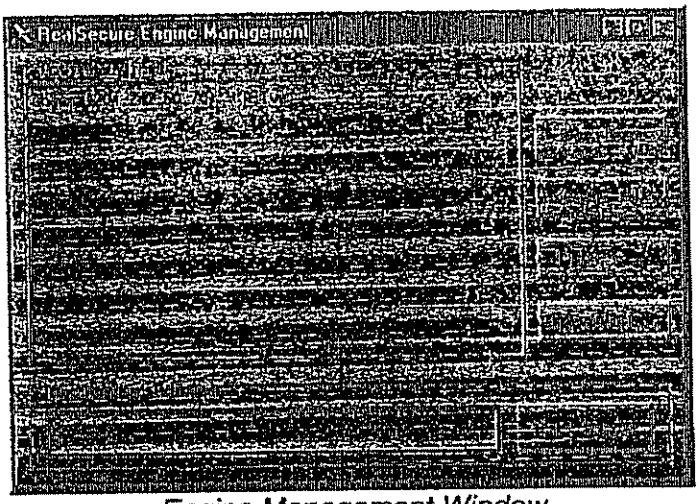
The filter configuration is defined to display all TCP traffic going through the network. Every feature is enabled in the features configuration.

Chapter 4:

Using RealSecure

OVERVIEW

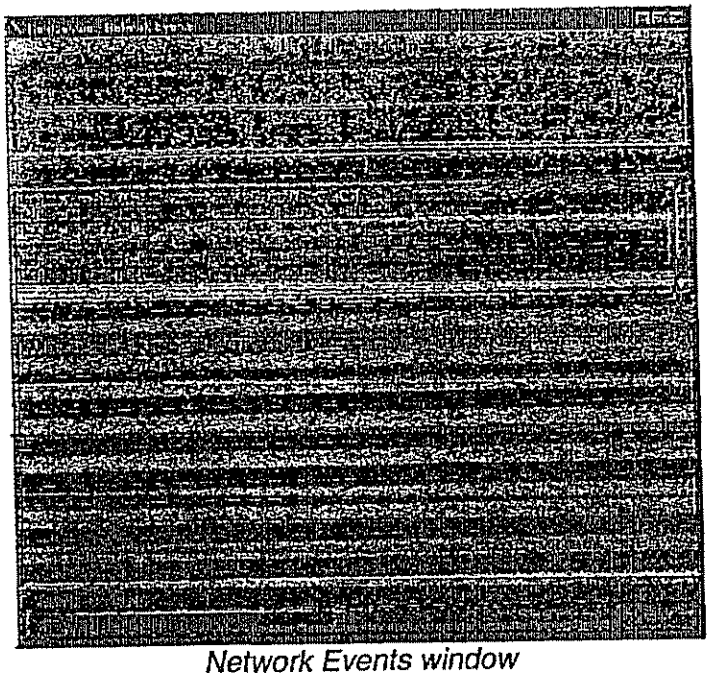
Upon selecting Configure Engines from the RealSecure toolbar, the Engine Management window displays. This window allows for starting, stopping and configuration of engines on each host. If engines are already running on the network, the GUI is automatically contacted and appears within a list on the Engine Management window. The list displays the active engines, one per line as follows...



Engine Management Window

To start an engine, type the name or IP address of a host on which you wish to run an engine and press Enter, or click on Start Engine. After a short delay, the engine will appear in the engine list and the Network Events window will pop up to start displaying events.

Remote engines may only be started on hosts that are configured for running sssd.



Network Events window

As events appear in the window, more information is provided about them by double-clicking on the entry.